**Amendments to the Specification:**

Please replace the paragraph beginning at page 67, line 6 of the Substitute Specification with the following replacement paragraph:

-- An array merging optimization may transform the data layout of arrays by merging the data of several arrays following the way they are accessed in a loop nest. This way, memory cache misses can be avoided. The layout of the arrays can be different for each loop nest. The example code for array merging presented below is an example of a cross-filter, where the accesses to array 'a' are interleaved with accesses to array 'b'. Fig. 23 illustrates a data layout of both arrays, where blocks of 'a' 2300 (the dark highlighted portions) are merged with blocks of 'b' 2302 (the lighter highlighted portions). Unused memory space 2304 is represented by the white portions. Thus, cache misses may be avoided as data blocks containing arrays 'a' and 'b' are loaded into the cache when getting data from memory. More details can be found in Daniela Genius & Sylvain Lelait, "A Case for Array Merging in Memory Hierarchies," *Proceedings of the 9th International Workshop on Compilers for Parallel Computers, CPC'01* (June 2001).--.